

AMENDMENTS TO THE SPECIFICATION

Please replace the paragraph beginning on page 2, line 11 with the following amended paragraph:

Contracts and many other types of agreements or documents are tightly integrated into many aspects of modern business. Companies often utilize such agreements to formalize the terms of a business relationship. For example, if a company is asked to perform a particular service for another company, a contract that specifies what types of services are to be performed, when those services are to be completed, and what the terms of payment are may be signed by both parties involved in the transaction. Creating the documents that ~~relates~~ relate to such business relationships is a complex process that often requires the involvement of attorneys or businessmen who are familiar with the details associated with the type of transaction or relationship being established. The involvement of such people usually requires a significant time investment from the people responsible for producing such a document. This can be prohibitively expensive in terms of the time or costs associated with involving the people having the appropriate expertise. However, the quality and value of the document that is ultimately generated largely depends upon the expertise of the documents drafter. A problem that exists during the conclusion of many different business deals is that the document drafter having the most expertise about the transaction to be concluded may not be available at the time the transaction is to be culminated. The problem increases if the document to be drafted carries any legal weight.

Please replace the paragraph beginning on page 3, line 14 with the following amended paragraph:

Existing systems provide a mechanism for generating contracts. For example, ContractMaker™, a product created by Digital Contracts, Inc. provides users with a way to generate contracts. The software contains a collection of numerous possible documents which subscribers can assemble interactively online. If the user has a valid subscription, the user is presented an interface for assembling the various part of the contract (e.g., steps 100 and 102). The interface enables the subscribing user to obtain access to a library of contract documents. Users may subscribe to different libraries in order to obtain contract documents covering different subject matter. The user may select from a variety of different contract types (e.g., step 104) and the system will then begin prompting the user for information to be filled into the contract. For example, current versions of the system may prompt the user for information about the parties to the contact (e.g., step 106). Once the parties are identified, the system presents a questionnaire to the user (e.g., step 108). The questionnaire embodies a logic tree which determines the questions, and their order and sequence. Questions are presented one-at-a-time as determined by the users answers to prior questions. Once the user completes all the questions in the logic tree (e.g., step 110) the system assembles the contract (e.g., step 112) and presents the drafted contract to the user. The user may edit the ~~contact~~ contract via a word processor and may change the assumptions or answers to questions by ~~back-tacking~~ back-tracking through the logic tree (e.g., via a wizard interface).

Please replace the paragraph beginning on page 4, line 17 with the following amended paragraph:

Although the ContractMaker™ is a good tool for creating contracts, it does not ~~provided~~ provide users with a flexible way to build a framework for generating various types of documents. Moreover, the system requires that the user build the document in sequence by traversing the logic tree associated with a particular contract document. Because of these and other limitations there is a need for a system that provides users with a more flexible way to generate documents.

Please replace the paragraph beginning on page 5, line 6 with the following amended paragraph:

In the foregoing discussion about current systems, the problems and limitations set forth as existent in the prior art are provided for ~~exemplarily~~ exemplary purposes. It should be clear to one of ordinary skill in the art that these problems also exist in other contexts or professions and that the invention may apply to situations other than the ones described herein.

Please replace the paragraph beginning on page 19, line 3 with the following amended paragraph:

The commission model is described in further detail in ~~pending patent application serial number 09/081857,~~ U. S. Patent No. 6,662,164, issued December 9, 2003, entitled "Method and Apparatus For Determining Commission", which ~~in~~ is incorporated herein by reference. Each compensation component may comprise some or all of the output generated by a commission engine or utilizing the commission model. The reader should note that in some instances the commission model generated by the commission engine is migrated into the system for generating

configurable documents using database migration techniques well ~~know~~ known to those of ordinary skill in the art. In other instances, the commission model is seamlessly integrated with the document generating system described herein.

Please replace the paragraph beginning on page 28, line 11 with the following amended paragraph:

The user may group components together and ~~specifies~~ specify how the group of components interrelate (e.g., Figure 2, step 212). Component-to-component relationships can be created between components within a document template. Relationships may be defined between a first set of components and a second set of components (e.g., compensation component X is related to textual component Y). When template processing is initiated (e.g., by the configuration engine) the relationship between the two sets is enforced on the components in the second set when all of the members of the first set of components are selected. In accordance with one embodiment of the invention, there are multiple types of relationships between components. For example, the user may associate each component with at least one of the following types of relationships: a requires, an includes, a can't work with (or excluded), and removes relationship. Once the relationship is defined the component may be categorized as a required component, optional component, or standard component. However, the designer may designate other rules that uniquely conform to the needs of the business for which the documents are generated.

Please replace the paragraph beginning on page 29, line 9 with the following amended paragraph:

Figure 3 shows groups of components the user has assembled. Each group comprises a document template with a set of components having defined relationships. Thus referring to Figure 3, if the user grouped the set of components that comprise document template 302 the user specifies at step 212 the relationship between compensation components 306, 308, and 310n and textual components 312 and 316n. The relationship definitions identify how each of the components may contribute value to a generated document. In accordance with one embodiment of the invention a relationship is established for each of the components that make up the document template. These relationships (also referred to as rules) are therefore defined and later applied to groups of components. The rule sets for each group may be customized to reflect the underlying business logic associated with the document to be generated. One embodiment of the invention enables the user to associate one or more relationships with each component. Figure 9 illustrates relationships between components according to an embodiment of the invention. All component types (e.g., standard, required, or optional) may include or exclude other components.

Please replace the paragraph beginning on page 38, line 2 with the following amended paragraph:

Figure 10 provides an example of classification performed by the configuration engine according to an embodiment of the invention. When the configuration engine begins processing it applies the rules defined by the user and assembles the components ~~need~~ needed to generate the requested document. An included component is a component that is included in a document by

default. For example, components 1004 and 1006 are automatically included in document 1002. For example, when a configuration user chooses the document template associated with document 1002, components 1004 and 1006 are automatically included in the configuration. A required choices classification specifies that a choice among a group of components must be made to create a valid document configuration. For example, components 1008, 1010, or 1012 must be selected to generate a valid document. An optional choice may or may not be included in the document generated by the configuration engine.

Please replace the paragraph beginning on page 40, line 1 with the following amended paragraph:

Figure 5 comprises a block diagram illustrating the how an embodiment of the invention provides a document template to the configuration engine which is configured to apply a rule set and thereby generate a configured document. Configuration engine 712 utilizes the internal representation (e.g., rules 500 which represents the various component relations) associated with each document template to generate document 500 based on user input. Thus, an embodiment of the invention provides the user with the ability to specify a particular document. Once the user indicates which document is to be produced, configuration engine 712 verifies the document specification to ensure the document conforms to the rule sets 500 previously input into the system. The document is then ~~provides~~ provided to the operating user so that it can be printed or otherwise utilized for its intended purpose.

Please replace the paragraph beginning on page 41, line 2 with the following amended paragraph:

~~Once~~ One implementation of the system designed in accordance with an embodiment of the invention relies on a set of agreements (Agreement objects), which are kinds of contracts. A selling agreement may detail the terms of compensation between a financial services company and a distributor that is selling its products. It is a contract that explains agreements established between financial services companies and their distributors. Typically, an agreement is with a distributor firm and includes a sales team hierarchy of the people that work for that firm. The flexibility of the system embodying the invention allows for a different commission model for each agreement.

Please replace the paragraph beginning on page 63, line 16 with the following amended paragraph:

With the system embodying the invention, the user can ~~eustomized~~ customize pre-defined commission models during the negotiation of an agreement. For example, perhaps the user ~~want~~ wants the starting value of a quota level to allow for changes within a specific range of values. DMS provides for this through [an] a Customization object. [An] A Customization object is a specification for an affected object (in this instance, the quota level). A contract component (ContractComponent) contains a collection of Customization objects. It is possible, through the use of customization objects, to allow for several customizations to be made to the commission model for a compensation component. The Customization class has two object properties for indicating what the affected object property is. One of these is called simply the AffectedObject, used to directly specify the affected object of the customization. The AffectedProperty is a string that specifies the property name on the affected object. DMS user interface employs the AffectedObject and AffectedProperty at run time to get and set the customized value. The other key Customization object property is called the OwningObject and is for locating the affected

object within the commission model (or whatever framework is being customized). The owning object has an object path associated with it. The path specifies how to get from the owning object to the affected object. These values are used in maintenance tools that manipulate the Customization object to know how the affected object relates to others in the commission framework.